REMARKS/ARGUMENTS

Claims 9-28 are pending in the application.

The amendment to claim 9 is intended only to correct a minor informality, and is not being made in response to a rejection.

The amendments to claims 19 and 26 are intended to avoid any potential problems concerning subject matter eligibility under §101 in light of recent cases. They are not intended to narrow the claims, and are not being made in response to a rejection.

The rejection under 35 U.S.C. §102(a) is respectfully traversed on grounds that U.S. Publication No. 2003/0028731 of Spiers et al. cannot anticipate any of claims 9-27. The rejection of claim 28 under 35 U.S.C. §103 is also traversed.

It is submitted that Spiers et al. discloses a network attached storage (NAS) device having an associated block storage device (for example, a hard drive). An NAS is a "device that is dedicated to data storage and has its own network address." Spiers et al. at para. 0012. The NAS comprises a network interface for communicating over LAN or WAN, an interface to the block data storage device, and a processor running an operating system. *Id.* at para 0013. The processor receives a "network command" and generates a string of one or more commands specific to the block data storage device that is necessary to carry out the network command. *Id.* The network command is sent by a host computer 16, in response to an application 38 requesting of operating system 40 to read or write a file. The operating system includes a special interface and driver called a "host block data storage processor" 42 that is specifically adapted to handle requests from the application program that involve the NAS's block data storage devices 16. *Id.* at para. 0038.

Therefore, in essence, the combination of the NAS device and the host computer's "virtual interface" and "virtual device" make a remote block storage device appear to an application on the host computer to be local, i.e. connected to the computer. Spiers et al describe this in paragraph 0015, as well as in paragraph 0038:

The host operating system 40 comprises a host remote block storage device processor 42 that is specifically adapted to handle requests from the application program 38 that involve the block data storage devices 26 associated with the NAS device 14. For illustration purposes, the block data storage devices 26 are represented within the host computer 16 as "virtual" block data storage devices 26'. Specifically, the host operating system 40 with the host remote block storage device processor 42 operates to: (1) receive requests from an application program that relate to the reading or writing of data to a file located on one of the block data storage devices 26, i.e., a file request; (2) translate the file request into one or more network commands; (3) cause a network command to be conveyed to the host network interface 36; (4) receive a response to a previously conveyed network command from the host network interface; and (5) transmit, if appropriate, the response to the application program.

Thus, in response to a request from the application for a file stored on a remote block data storage device the operating system generates a "network command" and transmits it over a network to a NAS, which then generates from the network command into one or more commands for operating a block data storage device connected to the NAS. *See, also, id.* at 0042. Consequently, it is submitted that, with this virtualization mechanism, a remote block data storage device – a hard drive – is made to look like a locally connected hard drive.

In the claimed subject matter, on the other hand, conventional write commands issued by an application to the host's computer system are being used, in effect, to tunnel device-specific commands, which are generated by an application, to a processor of a device that is connected to a computer. The application stores the command in a file, and asks the operating system of the host computer to write the file to the connected device. The connected device, pretending to be a storage medium by simulating one to the host computer, allows the file to be written to memory within the device. The claimed subject matter therefore has an advantage of not requiring use of a special device driver on the host computer for enabling an application to communicate with the attached device. By simulating conventional data storage connected to an interface – for example, by pretending to be flash memory connected at a USB port – the device is able communicate with the application using drivers that are already typically part of the operating system, and that are typically available to users without administrative privileges. See

specification at para. 0008 and 0009. Thus, the application generates the device-specific command, places it in a file, and asks the host computer to write it to the storage device.

Claims 9-18 and 28

Turning to claim 9, there are several errors in the reasoning supplied to show anticipation.

First, the examiner contends that "generating a device specific command by an application program on the computer" is met by paragraph 0013 of Spiers et al. However, there is no reference to an application program in this paragraph, much less one running on a processor of a host computer.

Second, the only "device specific command" that is referenced in this paragraph is a command generated by the NAS (using its block data storage processor 30) from a network command that it receives over a network from a host computer. It is not being generated by the application. The cited paragraph says:

> [0013] The NAS device of the present invention comprises two interfaces: (1) a network interface for transmitting/receiving communications to/from a network infrastructure (e.g., a LAN or WAN), where the network interface is capable of implementing a data packet protocol (e.g. TCP/IP) and (2) a block data storage device interface for transmitting/receiving communications to/from a block data storage device. The NAS device further comprises an operating system with a block data storage processor (which can be thought of as the target side of a new type of virtual device driver). The block data storage processor is capable of receiving a network command that has been received at the network interface and generating a string of one or more commands specific to the block data storage device that are needed to carry out the network command. For example, if the processor receives a "read" network command from the network interface, the processor may have to issue commands to a disk drive requesting the status and/or availability of the device, reserving the device, loading the read/write head onto the disk, seeking to the track at which a desired block of data is locating, and then searching for the sector on the track at which the block of data is located, and finally issuing the device specific read command to cause the requested data to be read. The block data storage processor is also capable of causing any such string of commands to be transmitted to the block data storage interface for subsequent conveyance to a block data storage device. Further, the processor is capable of

receiving any response to a command from the block data storage device interface and, if necessary, sending data to the network interface for transmission over a network infrastructure to the computer system that requested the data.

Third, there is not found, either in cited paragraph 0033 or any other part of Spiers et al. that the undersigned attorney can locate, any mention of a device specific command being stored in a special file. This paragraph states:

[0033]   The block storage device processor 30 is also capable of processing a network command to produce a string of one or more device specific commands for the particular one of the block data storage devices 26 to which a network command relates. For instance, if the network command is a "read" network command that is directed to a disk drive that is one of the block data storage devices 26, the block storage device processor 30 processor may issue device specific commands that reserve the drive, cause the actuator to move the read/write head to the track at which the block is located, read the block of data and put the data in a cache, and read the block of data from the cache. Typically, the second and subsequent device specific commands in a string are each generated after the outcome of the prior command is known, thereby allowing the processor to take into account, in determining the second and subsequent commands, the responses to one or more prior commands where there are multiple, possible outcomes to at least one prior command. For example, if the response to a device specific status command indicates that the device is busy, the processor may cause the status command to be reissued. If, however, the response to a device specific status command indicates that the device is available, the processor may issue a command to reserve the device. The block storage device processor 30 is further capable of conveying the device specific commands to the block storage device interface, which operates to convey the commands to the particular block data storage device. Additionally, the block storage device processor 30 is capable of receiving any replies to the string of commands from the block data storage devices 26 and processing any replies. The block storage device processor 30 is also capable of conveying any reply from the block data storage devices 26 that satisfies a network command to the storage device network interface 22. For instance, if the network command received at the network interface was a "read" network command, the processor 30 is capable of causing the block of data that has been read from the device to be conveyed on to the storage device network interface 22 for transport over the network infrastructure 18 to the host computer 16.

The cited paragraph appears to be discussing only the mechanism discussed above, which takes a request from an application, generates a network command, transmits the network command over a network to the NAS, which then translates it to commands for controlling the block data storage level device.

Fourth, there is no mention of a processor on an attached device executing the device-specific instructions stored in a special file written to the device in paragraph 0015, which the examiner cites as meeting this limitation. This paragraph discusses the operation of the <u>host</u> remote block storage device processor, which is element 42 in the drawings. In other words, it is talking about the host computer 16. Under the examiner's reasoning in connection with the other limitations of claim 9, at least as understood by applicant's representative, the examiner appears to be contending the network attached storage (NAS) device 14, not the host computer 16, is the "device" set out in claim 9. Furthermore, it is submitted that the NAS of Spiers et al. does not, in fact, appear to execute commands stored in files written to the NAS.

The rejections of dependent claims 10-18, as well as the rejection under §103 of dependent claim 28, rely on this same reasoning and therefore must be in error for at least the same reasons. In view of the errors, it appears unnecessary at this time to address errors in the additional reasoning supplied for each of these claims, but the right to address on appeal such errors is not waived and is respectfully reserved.

Claims 19-25

The examiner rejects independent claim 19 by incorporating by reference the reasoning of claim 9. Therefore it must contain at least the same errors as the reasoning supporting the rejection of claim 9. Given the errors, there appears to be no need to address the errors in the additional reasoning supplied to support the rejections of dependent claims 20-25. The right to complain about these errors is respectfully reserved and not waived.

Claims 26-27

The rejection independent claim 26, which is directed to just the device, is traversed on grounds similar to those given above.

First, the examiner cites paragraphs 0012, 0033, 0038, 0042, 0064 and 0067 as disclosing the limitation, "the device being specifically adapted for simulating a stored medium to an interfacing computer and receiving a special file containing a device specific command created by an application running on the computer and written to the device using the write command of the computer's operating system." However, as discussed above, the NAS 14 of Spiers et al. does not read a device-specific command from a special file written to it by a write command of the operating system of the host computer. It receives a network command and translates it into commands for the block storage device associated with the NAS.

Second, the examiner further contends that "file command" in paragraph 0042 of Spiers et al. is a "special file." But the "file command" references a command issued by an application on the host computer to the operating system, which then 'translates' it into a network command for transmission over a network to the NAS. Therefore, it cannot be the "special file" that a device reads, as the file command exists only on the host computer.

Third, the examiner cites paragraph 0067 and the last sentence in paragraph 0033 as meeting the limitation,

> the processor being adapted to read the device specific command in the special file in response to receiving the special file by means of the write command of the operating system of the computer through the interface with the computer, and to execute the device specific command contained in the special file.

The quoted passage of the examiner refers to the returning of the contents of the file on the associated block storage device by the NAS in response to receive a "read" network command. The block storage device processor 30 of the NAS does receive a network command to return a file, but it does not receive a file and then read from the file a device-specific command. It simply reacts to the command by generating, with the block data storage processor

30, device-specific commands issued to the block data storage device. A device-specific command is, under the examiner's reasoning used in connection with the rejection of the other claims and in the literal language of Spiers et al., the command generated by the block storage device processor 30 in response to receiving the network command. A network command cannot be both a special file and a device-specific command.

Therefore, in view of these errors, the examiner has not established a *prima facie* case of anticipation of claim 26 or of its dependent claim 27, which is rejected based on the same reasoning.

<u>Conclusion</u>

For these reasons, the rejection of claims 9-27 under §102, and of claim 28 under s §103, are in error. There are other errors in the reasoning, which, in view of those mentioned above, are unnecessary to discuss. The right to address those errors on appeal is not waived and is respectfully reserved.

Applicant respectfully requests entry of the above Amendment and reconsideration and allowance of the application. A failure to address any particular statement of the examiner should not be construed as acquiescence or acceptance of that statement, or a waiver of any right to complain on appeal of errors in the examiner's reasoning.

Applicant hereby authorizes the Commissioner to charge any fees due but not submitted with this paper to Deposit Account No. 07-0153. The Examiner is respectfully requested to call Applicant's Attorney for any reasons that would advance the current application to issue. Please reference Attorney Docket No. 125542-1005.

Respectfully submitted,

GARDERE WYNNE SEWELL LLP

_____/Marc A. Hubbard/_____
Marc A. Hubbard
Registration No. 32,506
ATTORNEY FOR APPLICANT

Date: January 28, 2009

3000 Thanksgiving Tower
1601 Elm Street
Dallas, Texas 75201-4761
(214) 999-4880 - Telephone
(214) 999-3880 - Facsimile

DALLAS 2012224v.1